UNITED STATES PATENT AND TRADEMARK OFFICE

amv

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/667,122	09/21/2000	William C. Moyer	SC11306TH	9170

23125 7590 05/17/2005

FREESCALE SEMICONDUCTOR, INC.
LAW DEPARTMENT
7700 WEST PARMER LANE MD:TX32/PL02
AUSTIN, TX 78729

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	**PAPER NUMBER**
2183	

DATE MAILED: 05/17/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

X

Office Action Summary	Application No.	Applicant(s)
	09/667,122	MOYER ET AL.
	Examiner	Art Unit
	David J. Huisman	2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 21 March 2005.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-25 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-25 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 21 September 2000 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-25 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Amendment as received on 3/21/2005.

Specification

3. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

Claim Objections

4. Claim 5 is objected to because of the following informalities: In line 4, replace "decode unit" with --decode control unit-- for increased clarity. Appropriate correction is required.

5. Claim 13 is objected to because of the following informalities: In line 4, replace "decode unit" with --decode control unit-- for increased clarity. Also, in lines 8-9, replace "a second sequence signal that" with --the second sequence signal--.. Appropriate correction is required.

6. Claim 18 is objected to because of the following informalities: In line 4, replace "decode unit" with --decode control unit-- for increased clarity. Appropriate correction is required.

Maintained Rejections

7. Applicant has failed to overcome the prior art rejections set forth in the previous Office Action. Consequently, these rejections are respectfully maintained by the examiner and are copied below for applicant's convenience.

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

9. Claims 1-9, 13-17, and 21-25 are rejected under 35 U.S.C. 102(b) as being anticipated by Hoyt et al., U.S. Patent No. 5,604,877 (herein referred to as Hoyt).

10. Referring to claim 1, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal

is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

11. Referring to claim 2, Hoyt has taught a processing unit as described in claim 1. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

12. Referring to claim 3, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) an execution unit which generates branch conditions and data addresses. See column 1, lines 44-47, and note that branch conditions are resolved in an execution stage. Therefore, it is

inherent that an execution unit exists to execute the branch instruction. Furthermore, it is further inherent that if a system loads and stores data to memory, as in Hoyt, then an execution unit must exists to execute load and store instructions such that data addresses for these operations are provided.

d) a decode control unit which decodes instructions. See Fig.3, component 60.

c) a fetch unit (Fig.3, components 30 and 35), coupled to the execution unit, the decode control unit, the address bus, and the data bus, for generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address.

See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

13. Referring to claim 4, Hoyt has taught a system as described in claim 3. Furthermore, it is inherent that Hoyt's decode control unit comprises an instruction register (IR). An IR is a known component which holds the instruction that is to be decoded and executed.

14. Referring to claim 5, Hoyt has taught a system as described in claim 3. Hoyt has further taught an address control unit, coupled to the decode unit and the execution unit, for receiving a branch condition signal and a branch decode signal and a load/store signal and for providing the first, second, and third sequence signals. See Fig.3 ands note that the address control unit may comprise any of the components 30, 40, 50, 70, and 80. These components are coupled to execution units 90, and decoder 60. All of the claimed signals are produced by these components (note the connections of components in Fig.3).

15. Referring to claim 6, Hoyt has taught a system as described in claim 5. Hoyt has further taught that the execution unit comprises a condition generator that provides the branch condition signal. See column 1, lines 44-47.

16. Referring to claim 7, Hoyt has taught a system as described in claim 6. Hoyt has further taught that the execution unit comprises a data address generator which provides a data address signal to the fetch unit. Clearly, if an execution unit is performing load instructions, then a data address signal must be provided to a fetch unit.

17. Referring to claim 8, Hoyt has taught a system as described in claim 7. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

18. Referring to claim 9, Hoyt has taught a system as described in claim 3. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

19. Referring to claim 13, Hoyt has taught a processor system comprising:

a) an execution unit. See Fig.3, component 90.

b) a decode control unit. See Fig.3, component 60.

c) a fetch unit, coupled to the execution unit and the decode unit, for providing addresses on an address bus which may be sequential and providing a first sequence signal and a second sequence signal for each address provided on the address bus wherein the first sequence signal indicates whether each address provided on the address bus may be sequential to an immediately preceding address on the bus and the second signal indicates whether each address provided on

the bus is sequential to the immediately preceding address on the bus. It should be noted from Hoyt's system that instructions are continuously fetched, and these instructions may include a branch instruction. Consequently, the current instruction address will inherently be sequential to the previous instruction address unless a taken branch instruction is encountered. Therefore, the first signal would be the BTB hit signal, as a hit denotes that a branch has been encountered and that the current address may be sequential to the previous address depending on the predicted direction of the branch. See column 8, lines 42-53. In addition, a second signal inherently is produced which indicates whether the program counter is to be incremented or whether it is to be replace by a predicted address. This second signal would be a signal that indicates whether the PC should be incremented (in case there's no branch) or if the PC is to be replaced (in case there is a branch, the PC needs to be replaced with the predicted branch address). When the second signal is in the first state, where the PC is to be incremented, since a branch does not occur, the address is sequential to the previous address.

20. Referring to claim 14, Hoyt has taught a system as described in claim 13. Hoyt has further taught that if the second sequence signal corresponding to one of the addresses indicates that the address is not sequential to the immediately preceding address, the first sequence signal corresponding to the address indicates that the address may not be sequential to the immediately preceding address prior to the second sequence signal indicating that the address is not sequential to the immediately preceding address. Clearly, the hit signal (first signal) in Hoyt must occur before the PC is updated for determining the next address. Consequently, if the second signal indicates that an address is not sequential (by being in the second state where a branch target is

written to the PC), then the first signal will have indicated a hit in the BTB prior to the second signal making its indication, as the hit must occur before prediction.

21. Referring to claim 15, Hoyt has taught a system as described in claim 14. Hoyt has further taught that the addresses may be instruction addresses, and wherein the fetch unit further provides a third sequential signal which indicates whether each address that is an instruction address is sequential to a previous instruction address. Note that a miss signal in the BTB would mean a branch instruction has not been encountered and consequently, the address is sequential to the previous address.

22. Referring to claim 16, Hoyt has taught a system as described in claim 15. Hoyt has further taught that the execution unit comprises a condition generator that provides a branch condition signal to the fetch unit. See column 1, lines 44-47.

23. Referring to claim 17, Hoyt has taught a system as described in claim 16. Hoyt has further taught that the decode control unit provides a branch decode signal and a load/store signal to the fetch unit. Clearly, all instructions are decoded, and in response to decoding a branch instruction, decoder 60 (Fig.3) will provide a branch decode signal. In addition, since loads/stores do occur in Hoyt (see column 5, lines 11-19), the load/store signals must be generated.

24. Referring to claim 21, Hoyt has taught a processing unit as described in claim 2. Hoyt has further taught that the second sequence signal is negated in response to resolving a branch condition code. See column 8, lines 59-65, and note that before the prediction can be provided, the type of branch must be determined. Therefore, the type field (condition code) of the branch

is checked to determine what branch condition is present (unconditional, conditional, return condition).

25. Referring to claim 22, Hoyt has taught a system as described in claim 14. Hoyt has further taught that the second sequence signal is negated in response to resolving a branch condition code. See column 8, lines 59-65, and note that before the second signal can be provided (the type of update to the PC), it must be determined if a branch is going to be predicted taken or not (condition code usually in the form of history bits). If so, the second signal will be provided to tell the system to write the predicted address to the PC.

26. Referring to claim 23, Hoyt has taught a processing unit as described in claim 2. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated and the first and second sequence signals are negated in a same clock cycle. It should be noted that the first sequence signal is always negated before the second sequence signal because there must be a hit (first signal) in the BTB before a prediction (second signal) can be provided. In addition, it is known that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a prediction will be made in that cycle, so that in cock cycle X+1, the predicted path may be fetched.

27. Referring to claim 24, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address.

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) an execution unit which generates branch conditions and data addresses. See column 1, lines 44-47, and note that branch conditions are resolved in an execution stage. Therefore, it is inherent that an execution unit exists to execute the branch instruction. Furthermore, it is further inherent that if a system loads and stores data to memory, as in Hoyt, then an execution unit must exists to execute load and store instructions such that data addresses for these operations are provided.

d) a decode control unit which decodes instructions. See Fig.3, component 60.

c) a fetch unit (Fig.3, components 30 and 35), coupled to the execution unit, the decode control unit, the address bus, and the data bus, for generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because

a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

f) wherein if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated and the first and second sequence signals are negated in a same clock cycle. It should be noted that the first sequence signal is always negated before the second sequence signal because there must be a hit (first signal) in the BTB before a prediction (second signal) can be provided. In addition, it is known

that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a prediction will be made in that cycle, so that in cock cycle X+1, the predicted path may be fetched.

28. Referring to claim 25, Hoyt has taught a system as described in claim 14. Hoyt has further taught that if the second sequence signal indicates that an address is not sequential to the immediately preceding address, the first signal indicates that the address may not be sequential to the immediately preceding address in a same clock cycle as the second sequence signal indicating that the address is not sequential to the immediately preceding address. It should be noted that the first sequence signal is always provided before the second sequence signal because there must be a hit (first signal) in the BTB before the PC is either incremented or replaced. In addition, it is known that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a PC update will be made in that cycle, so that in cock cycle X+1, the instruction from the address pointed to by the new PC may be fetched.

29. Claims 10-12 and 18-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Yoshida, U.S. Patent No. 6,205,536.

30. Referring to claim 10, Yoshida has taught a processing system for fetching instructions and data, comprising:

a) an address bus for providing a current address for retrieving a first instruction, a previous address for retrieving a second instruction, and a data address for retrieving data, wherein the data address occurs before the current address and after the previous address, and wherein the current address follows the previous address without any intervening addresses for retrieving instructions. See the abstract and Fig.27 and note that the third address provided is a data address, and it occurs between a previous and current address. Note that no intervening instruction addresses exist between instruction addresses 2 and 3.

b) a data bus for retrieving the first and second instructions and the data. This is an inherent component as data and instructions must be transmitted via some medium.

c) a fetch unit, coupled to the address bus and the data bus, for generating a first sequence signal that when asserted for the current address indicates that the current address is sequential to the previous address and when negated indicates that the current address may not be sequential to the previous address. Note from column 10, lines 30-36, that Yoshida has taught branch prediction for branch instructions, which would be a type of instruction represented in Fig.27. Even though Yoshida is silent as to how the branch prediction (first signal) is performed, it is inherent that when predicting branches, they may be predicted taken or not taken. Clearly, if a branch is predicted not taken, then the current address is sequential to the previous address. That is, the system begins fetching from the address immediately following the address of the branch instruction. On the other hand, if the branch is predicted taken, then the current address may not be sequential to the previous address. That is, the system begins fetching from the target address of the branch instruction. However, since the prediction won't be validated or invalidated until

the branch is finally executed, the fetching from the target is merely speculative, as opposed to known.

31. Referring to claim 11, Yoshida has taught a system as described in claim 10. Yoshida has further taught an address control unit for receiving a branch condition, a branch decode signal, and a load/store signal, and for providing the first sequence signal. Clearly, it is inherent that each of these signals is present. A branch condition must be realized in order to determine the direction of the branch. In addition, since all instructions are decoded, including branches, a branch decode signal will be provided. And, load/store signals are inherent in any system with registers, which are clearly references in Fig.8-10 (Rn). Any components which receive these signals are part of the address control unit. And, the first sequence signal is also provided by the address control unit.

32. Referring to claim 12, Yoshida has taught a system as described in claim 11. Yoshida has further taught:

a) an execution unit which provides the branch condition. See Fig.7, component 268, and note the condition code. Clearly, if the branch has to reference some condition in order to branch, then that condition must be produced by an execution unit. Usually, the condition is determined through operations such as addition.

b) a decode control unit which provides the branch decode signal and the load/store signal. See Fig.22, component 52, and note that since all instructions are decoded, signals corresponding to the instructions are produced by the decode unit.

33. Referring to claim 18, Yoshida has taught a processing system for fetching instructions and data, comprising:

a) an execution unit. See Fig.22, component 56.

b) a decode control unit. See Fig.22, component 52.

c) a fetch unit, coupled to the execution unit and the decode unit, for providing instruction and data addresses on an address bus and providing a first sequence signal that indicates whether each instruction address provided on the address bus is sequential to an immediately preceding instruction address even if a data address is provided between the instruction address and the immediately preceding instruction address. Clearly, since Yoshida's system executes branches (see Fig.7), it must be determined if every instruction address is sequential to its preceding address. That is, every address will either be sequential to its preceding address or it will not be sequential due to a branch. Therefore, the first signal is merely any signal which dictates that a prediction is or is not to be used (and note from column 10, lines 30-36, that prediction is employed). For instance, a first signal exists which would cause the system to either use a predicted address for a branch (which could be non-sequential) or to use an incremented value of the program counter, a component which inherently exists. Furthermore, it can be seen from Fig.27 that data addresses may occur between instruction addresses. It should be noted that even if a data address occurs between two instruction addresses, the system must still know whether the current instruction address is sequential or not sequential (whether to choose the next or a possible non-sequential prediction) to the previous instruction address, thereby requiring the existence of the first signal.

34. Referring to claim 19, Yoshida has taught a unit as described in claim 18. Yoshida has further taught that the execution unit comprises a condition generator that provides a branch condition signal to the fetch unit. See Fig.7, component 268, and note the condition code.

Clearly, if the branch has to reference some condition in order to branch, then that condition must be produced by an execution unit. Usually, the condition is determined through operations such as addition.

35. Referring to claim 20, Hoyt in view of Yoshida has taught a unit as described in claim 19. Yoshida has further taught that the decode control unit provides a branch decode signal and a load/store signal to the fetch unit. See Fig.22, component 52, and note that since all instructions are decoded, signals corresponding to the instructions are produced by the decode unit.

Response to Arguments

36. Applicant's arguments filed on March 21, 2005, have been fully considered but they are not persuasive.

37. Applicant argues the novelty/rejection of claims 1, 3, 13, and 24 on pages 11-14 of the remarks, in substance that:

> "Hoyt is not generating a first sequence signal as claimed. Instead, Hoyt is teaching a way of resolving a return from subroutine instruction using the BTB to determine if there is an upcoming branch instruction in the BTB. Hoyt uses this information to make a branch prediction. Importantly, however, Hoyt <u>never</u> teaches generating a signal using this information. Hoyt derives information from a BTB to determine if a hit has occurred. Also, Hoyt is comparing the current address to a subsequent address when making the determination. In sharp contrast, applicants claim comparing the current address with a previous address to generate the signal."
>
> "...the Office Action asserts that Hoyt teaches applicants' limitation of "generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address." Again, the Office Action's reliance on Hoyt for such teaching is misplaced. Hoyt is describing a well-known method of using branch history to predict whether a branch should be taken...Hoyt does not teach or suggest generating any signals when the branch outcome decision is made."
>
> Regarding the third sequence signal, "Hoyt is describing the situation of a mispredicted branch...and simply describes that, when a branch has mispredicted, the instructions that have been fetched from the instruction pipeline must be flushed. This has nothing to do whatsoever with applicants' claim limitation....Not only does Hoyt not teach or suggest generating a "signal" when the current address is not sequential to the previous address, but by the time a misprediction occurs the address with the branch that was mistakenly taken has long been processed...The examiner tries to redefine the previous address as the branch instruction

> address and the current address as the address from which to fetch after the previous address. However, this redefinition simply won't work."

38. These arguments are not found persuasive for the following reasons:

a) Regarding the first argument, as discussed in the rejection of claim 1, for instance, the first sequence signal is the BTB "hit" signal. When a previous address (from which an instruction is already fetched) causes a hit in the BTB, this means that a current address (from which the processor is now concerned with) may or may not be sequential to the previous address. The "hit" means a branch prediction is to be made. If the prediction is "not taken", then the current address will be sequential to the previous address (this represents fall-through). If the prediction is "taken", then the current address will not be sequential to the previous address (this represents branching to the branch target). As can be seen, the "hit" signal indicates a current address may not be sequential to the previous address. Whether it actually is or not depends on the prediction.

b) Regarding the second argument, as discussed in the rejection of claim 1, for instance, the second sequence signal is the branch prediction itself. The signal could either represent a "taken" or "not taken" prediction based on the history bits in the BTB. When the signal specifies taken, then that means the branch is to jump to some non-sequential address, thereby signaling to the system that the current address is not sequential to the previous address. It should be realized that everything that causes an action in a computer system may be construed as a signal, as actions only occur in response to some signal. Some signal in the system must exist which specifies taken and not-taken predictions; otherwise, how would the predictions ever be detected?

c) Regarding the third argument, as discussed in the rejection of claim 1, the third sequence signal is the branch outcome signal, which determines if a misprediction has occurred or not.

That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Therefore, the examiner believes this feature of Hoyt is very relevant to applicant's claimed invention. In addition, no redefinition of the previous and current addresses ever took place. On the contrary, the examiner's interpretation of the addresses was consistent through all of claim 1, for instance. The previous address is the address belonging to the branch instruction whereas the current address is the address from which to fetch an instruction immediately after the previous address. It should be noted that the current address is not fixed at a particular value, but instead, it represents any address which is to be fetched from after the previous address (so it is the sequential and/or target addresses of the branch). For instance, assume you have a branch at address 0000 which has a target address of 1111. The previous address is 0000 and if the branch is predicted taken, the current address is 1111, as this is the correct address to fetch from at this point in time. However, if a misprediction occurs shortly thereafter, then the current address is not 1111. Instead, the current address is 0001, as this is the correct address to fetch from at this point in time. Note that in either case, both addresses (0001 and 1111) may be considered "the current address" because both contain instructions which, based on the circumstance, are to be fetched immediately after fetching an instruction from the previous address. In the case where 0001 is the becomes the current address after an incorrect taken prediction, address 0001 is immediately following the previous address as far as fetching is

concerned. Address 1111 was incorrect and therefore, the system is to act as if instructions were never fetched from that location.

39. Applicant argues the novelty/rejection of claim 13 on page 14 of the remarks, in substance that:

> "In Hoyt, if a hit is encountered in the BTB, the address is not sequential to the previous address because the BTB is a table of branch addresses and, by definition, the "branch to" address will not be sequential to the current address."
>
> "Applicants respectfully disagree with the examiner and submit that Hoyt does not teach a second signal that is inherently generated. Instead, as the Office Action points out, a program counter (PC) is either incremented or is loaded with a branch address. When the PC is loaded with either an incremented address or a "branch to" address it is simply updated with no signal lines used to send a signal that the PC is either sequential or is not sequential to a previously loaded address. Hoyt simply does not teach or suggest sending or generating a signal indicating whether the address in the PC is sequential to the previous address that was in the PC."

40. These arguments are not found persuasive for the following reasons:

a) Regarding the first argument, as discussed in the rejection of claim 13, the first sequence signal is the BTB "hit" signal. This signal indicates that there is prediction information available for the corresponding branch instruction. However, this does not automatically mean that the current address is not sequential to the previous address. The examiner agrees that if the branch is taken (or predicted taken), then the current address is not sequential to the previous address. However, if the branch is predicted as "not taken", then the current address is the fall-through address (or sequential address). As is known in the art, taken and not-taken predictions are made based on history bits, which are used by the BTB of Hoyt. If the history indicates that it is more beneficial to predict "taken", then that is what will be done. However, if the history indicates that it is more beneficial to predict "not-taken", then that is what will be done.

b) Regarding the second argument, as discussed in the rejection of claim 13, the second signal inherently exists so that the program counter (PC) may be modified appropriately. Applicant, from the above argument, appears to agree that two different types of addresses may be loaded into the PC; either a target address associated with a branch instruction, or a sequential address, which is merely an incremented version of the previous address. If there are two possible addresses to load into the PC, then a signal must inherently exist in order to determine which of the addresses will be loaded. The loading of one or the other address does not happen by itself. Some action is taken in the system which causes one or the other address to be loaded. It should be realized that everything that causes an action in a computer system may be construed as a signal, as actions only occur in response to some signal.

41. Applicant argues the novelty/rejection of claims 10 and 18 on pages 16-19 of the remarks, in substance that:

> "The Office Action contends that Yoshida teaches generating a sequence signal. Upon closer inspection of Yoshida, however, it is clear that Yoshida never teaches or suggests generating or using a sequence signal...The fact that the Office Action admits that Yoshida is silent as to how the branch prediction is performed is a telltale sign that Yoshida does not teach or suggest providing a signal that indicates whether addresses are sequential to one another." Applicant further points out that none of the signals shown in Fig.27 of Yoshida read on applicant's claimed signal and that upon an electronic search of Yoshida, no mention of "sequential instructions," "sequential address," "sequential signal," or and other "sequential" or "sequence" term was ever found which would support the Office Action.

42. These arguments are not found persuasive for the following reasons:

a) As discussed in the rejection of claim 10, for instance, Yoshida has taught branch prediction and that when predicting branches, prediction options inherently include "taken" and "not-taken". The Office Action stated that Yoshida was silent as to the implementation of the prediction function not because Yoshida does not teach sequence signals, but because Yoshida

does not explicitly teach these signals. As the examiner pointed out though, the branch prediction signals are inherent. More specifically, when predicting a branch, if the branch is predicted taken, then the current address is predicted as being not sequential to the previous address, whereas if the branch is predicted not-taken, then the current address is predicted as being sequential to the previous address (see Hoyt for an example of how this is done). The prediction itself is inherently the first sequence signal which specifies whether or not the current address is sequential to the previous address. This signal exists because branch prediction exists, and the system must be informed as to what the next address will be (for fetching and execution purposes). Again, it should be realized that everything that causes an action in a computer system may be construed as a signal, as actions only occur in response to some signal.

Conclusion

43. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
May 11, 2005

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100